

VC++ C++ C# Algorithm

[C++博客](#) [首页](#) [新随笔](#) [联系](#) [聚合](#) [管理](#) 21 Posts :: 3 Stories :: 31 Comments :: 0 Trackbacks

常用链接

[我的随笔](#)
[我的评论](#)
[我参与的随笔](#)

留言簿(4)

[给我留言](#)
[查看公开留言](#)
[查看私人留言](#)

随笔档案(21)

[2007年3月 \(13\)](#)
[2007年1月 \(1\)](#)
[2006年12月 \(5\)](#)
[2006年11月 \(2\)](#)

文章档案(3)

[2007年1月 \(2\)](#)
[2006年12月 \(1\)](#)

相册

[照片](#)

资料网站

[C++之父主页](#)
[题库吧](#)
[数据库面试题|Java面试题|C++面试题|.NET面试题](#)
[微软在线课程](#)

搜索

最新评论 [XML](#)
[1. re: c++名字查找](#)

 是
 --周敏桢

[2. re: c++名字查找](#)

 名字区码
 --王

[3. re: SDL游戏编程\(3\)使用SDL扩展类库显示PNG图片和字体](#)

 评论内容较长,点击标题查看
 --dnnit

[4. re: c++名字查找](#)

 桔柑
 --黄建宁

[5. re: c++名字查找](#)

 我要找到王疆
 --王疆

评论排行榜

[1. c++名字查找\(13\)](#)

SDL游戏编程(7)键盘事件

这次我们看看如何在SDL中处理键盘事件，每次事件发生以后，有关事件的所有信息都存储在SDL_Event类型的变量当中，查查手册我们可以知道，实际SDL_Event是一个联合体。

```
typedef union{
    Uint8 type;
    SDL_ActiveEvent active;
    SDL_KeyboardEvent key;
    SDL_MouseMotionEvent motion;
    SDL_MouseButtonEvent button;
    SDL_JoyAxisEvent jaxis;
    SDL_JoyBallEvent jball;
    SDL_JoyHatEvent jhat;
    SDL_JoyButtonEvent jbutton;
    SDL_ResizeEvent resize;
    SDL_ExposeEvent expose;
    SDL_QuitEvent quit;
    SDL_UserEvent user;
    SDL_SywWMEvent syswm;
} SDL_Event;
```

当发生键盘事件时，key变量就是有效的。

```
typedef struct{
    Uint8 type;
    Uint8 state;
    SDL_keysym keysym;
} SDL_KeyboardEvent;
```

key变量是一个结构体，其中keysym成员包含了按键信息。

```
typedef struct{
    Uint8 scancode;
    SDLKey sym;
    SDLMod mod;
    Uint16 unicode;
} SDL_keysym;
```

在keysym的成员当中，sym记录按键所对应的虚拟键。

比如说向上就是SDLK_UP；向下就是SDLK_DOWN；大家可以自己去查手册。

下面就看一个处理键盘消息的实例吧。

该例子中，我们一旦发现有上下左右四个键被按下，就马上在屏幕上显示相对应的消息，这里用到了扩展类库SDL_TTF，如有不清楚的，参考一下前面的几篇文章。

```
#include "SDL.h"
#include "SDL_ttf.h"
SDL_Surface *screen=NULL;
SDL_Surface *up=NULL;
SDL_Surface *down=NULL;
SDL_Surface *left=NULL;
SDL_Surface *right=NULL;
SDL_Surface *message=NULL;
TTF_Font *font=NULL;
//screen to show on window
```

- 2. 循环的效率(4)
- 3. SDL游戏编程(3)使用SDL扩展类库显示PNG图片和字体(4)
- 4. SDL游戏编程(8)鼠标事件(3)
- 5. 一道c++的面试题目(2)

```

const int SCREEN_BPP=32;
SDL_Color textColor={255,255,255};

int main( int argc, char* args[] )
{
    //Start SDL
    bool quit=false;
    SDL_Init( SDL_INIT_EVERYTHING );
    if(TTF_Init() == -1)
        return false;

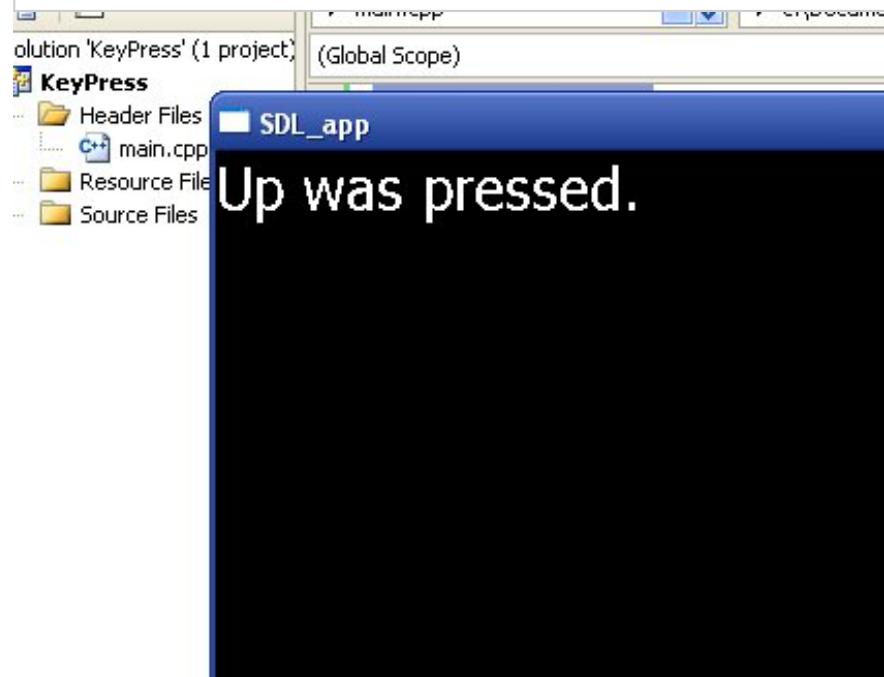
    screen = SDL_SetVideoMode( 600, 480, SCREEN_BPP, SDL_SWSURFACE );
    if(screen==NULL)
        return false;
    font=TTF_OpenFont("tahoma.ttf",28);
    up = TTF_RenderText_Solid( font, "Up was pressed.", textColor );
    down = TTF_RenderText_Solid( font, "Down was pressed.", textColor );
    left = TTF_RenderText_Solid( font, "Left was pressed.", textColor );
    right = TTF_RenderText_Solid( font, "Right was pressed", textColor );
    SDL_Event event;
    while(!quit)
    {
        if(SDL_PollEvent(&event))
        {
            if(event.type == SDL_KEYDOWN)
            {
                switch(event.key.keysym.sym)
                {
                    case SDLK_UP: message=up;break;
                    case SDLK_DOWN: message=down;break;
                    case SDLK_LEFT: message=left;break;
                    case SDLK_RIGHT: message=right;break;
                }
            }
            if(event.type == SDL_QUIT)
                quit=true;
        }
        if(message != NULL)
        {
            Uint32 colorVal=SDL_MapRGB(screen->format,0,0,0);
            SDL_FillRect(screen, &screen->clip_rect,colorVal);
            SDL_BlitSurface(message,NULL,screen,NULL);

            message=NULL;
        }
        if(SDL_Flip(screen) == -1)
        {
            return false;
        }
    }
    //Quit SDL
    SDL_Quit();

    return 0;
}

```

L}



posted on 2007-03-12 19:27 大熊猫 阅读(833) 评论(0) 编辑 收藏 引用

找优秀程序员，就在博客园

标题

re: SDL游戏编程(7)键盘事件

姓名

主页

验证码

 * 8658

内容(提交失败后,可以通过“恢复上次提交”恢复刚刚提交的内容)

 Remember Me?[登录](#) [使用高级评论](#) [新用户注册](#) [返回页首](#) [恢复上次提交](#)

[使用Ctrl+Enter键可以直接提交]

[【推荐】超50万行VC++源码: 大型组态工控、电力仿真CAD与GIS源码库](#)

Gmail au bureau

Une adresse e-mail
professionnelle et 30 Go
de stockage, maintenant.



网站导航: [博客园](#) [IT新闻](#) [BlogJava](#) [知识库](#) [程序员招聘](#) [管理](#)

Powered by:

[C++博客](#)

Copyright © 大熊猫